J.King Saud Univ., Vol. 2, *Eng. Sci.* (2), pp. 295–308 (1410H/1990)

Simulation of Isarithmic Congestion Avoidance in Computer Networks

Mohammad Ilyas
Department of Computer Engineering, Florida Atlantic University
Boca Raton, Florida 33431

Abstract. This paper deals with global (also called isarithmic) congestion avoidance in computer communication networks. In this technique, data traffic inside a computer network is kept to a comfortable level by imposing an upper limit on the number of data packets that can be present in the network at any time. This technique is implemented by using permits. A message seeking entry into the network must acquire a permit and a message leaving the network must release the permit. The paper shows that performance of a computer network using isarithmic congestion avoidance, depends upon the distribution of permits among the network nodes. The paper considers three schemes for distribution of permits: complete sharing (CS), complete partitioning (CP), and proportional sharing (PS). The main idea behind these schemes is to prevent active users from monopolizing all the permits and also to prevent relatively inactive users from starving for permits. This paper presents a comparison of these schemes by simulation a ten-node computer network. Simulation results are presented in terms of average message delay and throughput. It has been observed that the proportional sharing scheme gives the best overall performance.

Introduction

Computer communication networks cannot afford to accept all the data traffic that is offered to them. In the absence of an effective traffic flow control mechanism, these networks may start experiencing congestion as the offered data traffic increases. As a computer networks moves towards a congestion state, its throughput (average number of messages handled successfully per unit time) degrades and the average network transit delay per message increases. This situation may lead a computer network to a deadlock (a chronic form of congestion) and the data traffic in the network may become stand-still [1]. Therefore, the choice of an effective method of congestion avoidance has always been and still is an important consideration in the design of a computer communication network.

A congestion avoidance algorithm is simply a "code of ethics" which ensures that a network's resources (link bandwidth and buffers etc.) are not being abused. Abuse of resources means their unfair allocation to the network users. In such a situation, some users tend to monopolize all the resources while other users remain deprived of their fair share of the resources. As long as congestion avoidance rules are obediently observed by the network users, there are less chances that any unpleasant congestion effects would be observed. However, if these rules are not respected by the users, problem such as network congestion, blocking of buffers and deadlocks may be experienced. The code of ethics on algorithms that try to maintain a smooth and trouble-free flow of data traffic in computer communication networks, are commonly referred to as flow control or congestion avoidance methods [1,2,3].

Congestion in computer communication networks occurs when the rate of data traffic inflow exceeds the service rate of the network. An intuitively reasonable method of avoiding network congestion is to limit the number of data packets allowed to travel through a network at a time. An implementation of this method is referred to as global or isarithmic congestion avoidance [1,4,5].

A computer communication network operating on global or isarithmic principle of congestion avoidance, is provided with a fixed number of permits (also called tokens) which are randomly distributed among the network nodes. A message seeking entry into the network must acquire a permit before it can be allowed to travel through the network. At its destination node, the message must release the permit before leaving the network. As every message needs a permit to travel through the network, the total number of messages inside the network cannot exceed the total number of permits. Therefore, the traffic inside the network remains under control. However, the performance (in terms of throughput and average message delay) of a computer network depends upon the distribution of permits among the nodes of the network. In this paper, we consider three schemes for distribution of permits among the nodes of a computer network and observe the effects of these schemes on the throughput and delay performance of the network. The performance is evaluated by simulating a ten-node network.

Permit Distribution Schemes

This section describes three schemes for distribution of permits among a network's nodes. These schemes are complete sharing (CS), complete partitioning (CP) and proportional sharing (PS). All these schemes are implemented by using two quantities MIN(i) and MAX(i). MIN(i) is the number of permits allocated to node i, whereas MAX(i) is the maximum number of permits that node i can use at a time. A total number of permits to be distributed among the network nodes is denoted by

TNP. It has been assumed that a network has N source nodes and the traffic entering from node i is TR(i) where i=1,2,3....,N. Therefore, the total traffic (TRT) entering the network can be written as TRT = TR(1) + TR (2) + TR (3) + ... + TR(N). A brief description of these schemes follows.

Complete Sharing (CS)

This scheme is the general permit distribution scheme associated with global or isarithmic congestion avoidance method [3]. In this scheme, all the permits are (in a sense) placed in a pool. Each source node can request a permit as and when needed. There is no pre-allocation of permits to the network nodes. Therefore, MIN(i)=0 for i=1,2,3,...,N. In addition, there is no upper limit on how many permits a node can acquire at a time. However, no node can possibly acquire permits more than the quantity of permits available inside the network. Therefore, MAX(i)=TNP for i=1,2,3,....,N. In this scheme there is a potential problem of certain active nodes monopolizing all the permits and leaving no permits for relatively less active nodes.

Complete Partitioning (CP)

In this scheme each source node is provided with an equal number of permits. No node is allowed to share with or request permits from other nodes. Therefore, in this scheme MIN(i)=MAX(i)=TNP/N. This scheme avoids the problem of active nodes having monopoly over all the permits and relatively less active nodes becoming starved of permits. However, active and inactive nodes are given an equal number of permits, which is not fair. If all nodes have the same rate of traffic inflow, then this scheme should work very well but this is not the case in real networks.

Proportional Sharing (PS)

In this scheme permits are allocated to source nodes proportional to the rate of traffic inflow. Therefore, MIN(i)=MAX(i)=(TR(i)/TRT)*TNP. This scheme does not allow source nodes to share with or request permits from other nodes. This scheme does not only resolve the issue of active nodes dominating the use of permits and leaving nothing for relatively less active nodes, but is also fair at the same time. As will be discussed later, this scheme has shown better performance especially when traffic rate is high.

Simulation Model and Assumptions

The model used for simulating the global congestion avoidance schemes is shown in Fig. 1. The model consists of ten nodes and its end-to-end traffic matrix is shown in Fig. 2. A "*" at entry (i,j) means that source node "i" sends data messages to destination node "j". The data traffic rates on individual links can be calculated with the help of the routing matrix and the end-to-end traffic matrix. The routing matrix for this network is shown in Fig. 3. In the routing matrix, entry (i,j) indicates the next node on the route from source node "i" to destination node "j". The restrictions on data traffic inflow are exercised at source node by implementing an appropriate congestion avoidance scheme. It has been assumed that the permits can be made available instantaneously on request. It is also assumed that the storage capacity at the network is very large and that the network is operating in steady state conditions. Message lengths and hence their service times are assumed to be exponentially distributed and messages arrive at source nodes according to a Poisson process [6]. All messages are assumed to be single-packet messages and the transmission channels in the network are assumed to be noise-free.

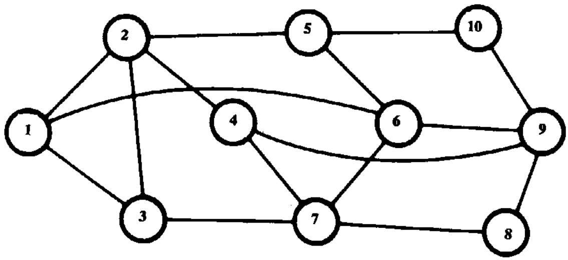

Fig. 1. The simulation model

S \ D	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	*	*	*	0	0	0
2	0	0	*	0	0	0	*	0	0	0
3	0	*	0	0	0	0	0	0	0	0
4	0	0	0	0	0	*	0	0	0	0
5	*	0	0	0	0	0	*	0	0	*
6	*	0	0	*	0	0	0	0	0	0
7	*	*	0	0	*	0	0	0	*	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	*	0	0	*
10	0	0	0	0	*	0	0	0	*	0

Fig. 2. Network traffic matrix

S \ D	1	2	3	4	5	6	7	8	9	10
1	-	2	3	2	2	6	3	6	6	2
2	1	-	3	4	5	5	4	4	4	5
3	1	2	-	2	2	1	7	7	2	2
4	2	2	2	-	2	9	7	9	9	9
5	2	2	2	2	-	6	6	10	10	10
6	1	5	1	9	5	-	7	9	9	9
7	3	4	3	4	6	6	-	8	8	8
8	7	9	7	9	9	9	7	-	9	9
9	4	4	4	4	10	6	8	8	-	10
10	5	5	5	9	5	9	9	9	9	-

Fig. 3. Traffic routing matrix

Simulation Program

The simulation program used in this study is written in FORTRAN and is executed on a VAX 11/780 computer. In developing this simulation program, we have used an efficient approach called the event-scheduling [7,8]. In this approach, the simulation clock is advanced by a variable quantity that represents the time interval between the current event and the next event to take place [7,9]. The simulation program uses the same principle as used in a simulation package, SIMNET, developed by the author [7]. The flow chart of the simulation program is shown in Fig. 4.

The program starts by reading the input parameters that are provided by the user at the outset of the program. The input parameters include the average traffic rates, the total number of permits, the average message service time, the type of congestion avoidance scheme to be implemented and some control parameters so as to stop the simulation program. The program progresses with by executing events in a sequential order. After every event, the program checks whether or not sufficient simulation time has elapsed or a sufficient number of messages have passed through the network. If not, the next sequential event is picked up and executed. This process continues until enough simulation has elapsed or enough messages have passed through the network. If either of these conditions has been met, the program stops after calculating the output parameters.

The program consists of a main program and three subroutines. These subroutines are INIL, ARRV and DPTR. These subroutines work equally well for all

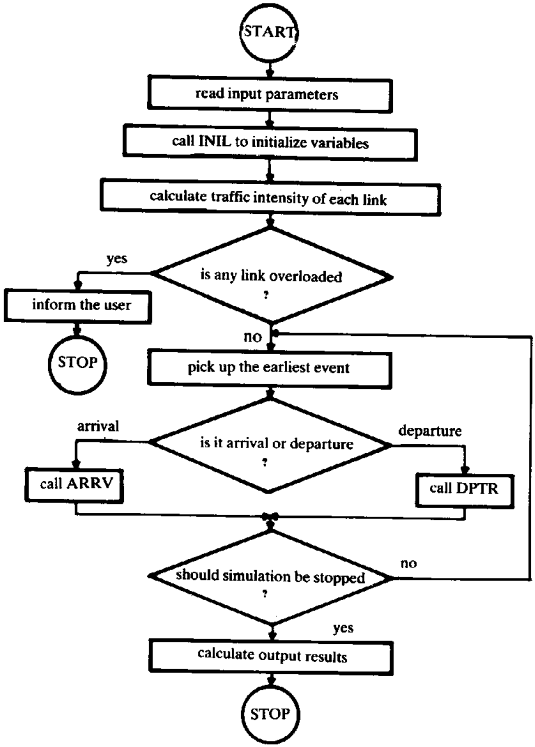

Fig. 4. Flow chart of simulation program

the three permit distribution schemes discussed in this paper. A brief description of these subroutines is as follows:

Subroutine INIL

This subroutine is called at the beginning of the main (or the calling) program. The subroutine initializes all program values to their appropriate starting values. This is done on the basis of the input parameters provided by the user at the beginning of the simulation program. For instance, all queue sizes are initilized to zero, minimum and maximum limits on the number of permit for each source node are initialized to their appropriate values depending upon the permit distribution schemes, etc.

Subroutine ARRV

This subroutine updates the network state when an external arrival occurs at a source node. If the queue size at that was zero prior to this arrival, then the departure of the newly arrived message is scheduled. Otherwise, this message waits in the queue to depart at its turn. Before the subroutine returns the control to the main program, it schedules the next arrival only if the global flow control restrictions allow an arrival to take place. Otherwise, arrivals at this node are disabled until they are allowed.

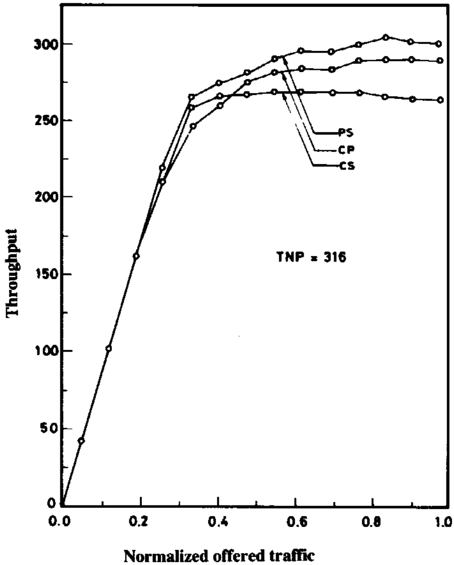

Fig. 5. Network throughput versus normalized offered traffic

Subroutine DPTR

This subroutine is called to update the network state when a departure occurs from a node. The departure could be from an intermediate node or it could be from a destination node. If the departure is not from a destination node, the subroutine simply pushes the queue forward at the departing node, and places, the departing message at the next node's queue. However, if a message departs from a destination node, the subroutine calculates transit delay of the message, updates the number of messages delivered and updates the delay histograms. In either case, the subroutine schedules the next departure, if any, before returning the control to the calling program.

Simulation Results

This section deals with the simulation results for the global congestion avoidance schemes discussed in section II. For all these results, it has been assumed that the message lengths are exponentially distributed with an average of 1200 bits and the network channel capacities are equal to 50 Kbps each. For each point in simulation results, we ran simulation for more than 10,000 messages so as to obtain converged results. Confidence interval were too small to be plotted.

Fig. 5 shows the network throughput versus the normalized offered traffic. It can be seen that for a low value of traffic load, the performance of all schemes is the same because there is no contention for permits. However, as the traffic load increases, the contention for permits also increases and different permit distribution schemes yield different performance. It can also be observed that the performance of the PS scheme is the best among all. The lowest throughput is for the CS scheme and that of the CP scheme lies in between. Apparently the performance of a system heavily depends upon how fairly users are allowed to share the resources. The CS scheme is absolutely unfair because in this scheme highly active users may use up all the permits and others may have to wait for a long period of time before they can get hold of a permit. The CP scheme offers a relatively better fairness. However, this scheme is too hard on active users and too generous towards relatively less active users. This is because all users are allocated the same amount of permits. The PS scheme, however, allocates permits keeping in view that how active a user is. In this scheme permits are allocated to a node proportional to the data traffic entering the node. This is a very fair scheme. That is why the throughput performance of the PS scheme is better than that of the CP and CS schemes.

Fig. 6 shows the network transit delay per message versus the normalized offered traffic load. As can be seen the highest delay is for the CS scheme and the lowest delay is for the CP scheme. The PS scheme offers the network transit delay which is less than that of the CS scheme but is greater than that of the CP scheme. An intuitive explanation for this is that in the CS scheme, active users tend to monopolize the permits. Therefore, the nodal queues start building up at some of the intermediate nodes that are being used by active users and contribute heavily to the network transit delay of a message. In the CP scheme, active users are severely curbed down and there is a mixture of traffic entering the network from all users. Therefore, all the network nodes take part in handling the network traffic and queues do not build only at a few of the nodes. Hence, the network transit delay per message remains reasonably low. The PS scheme is not as tight in applying restrictions on data traffic inflow as the CP scheme and not as loose as the CS scheme. Therefore, the network transit delay per message in the PS schemes less than that in the CS scheme and larger than that in the CP scheme.

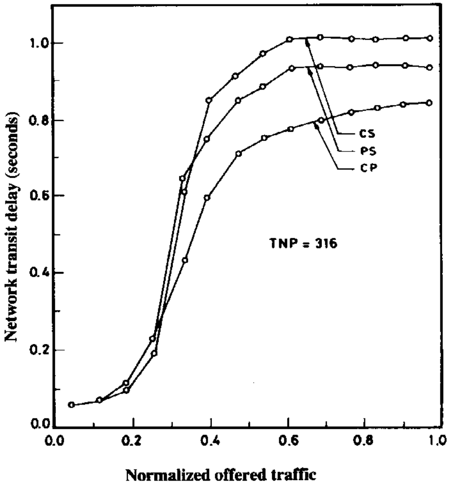

Fig. 6. Network transit delay per message versus normalized offered traffic

Fig. 7 and 8 show the network throughput and the network transit delay, respectively, versus the total number of permits for a fixed value of the normalized offered traffic load. The throughput increases gradually with the increase in the number of permits and as the total number of permits is increased further, the throughput levels off. The throughput of the PS scheme is the lowest in the beginning but as the number of permits is increased, it becomes the highest. The opposite is true for the CS scheme. It appears from these curves that the throughput performance of a computer network not only depends upon that which permit distribution scheme is being used but also depends upon the total number of permits. The network transit delay per message also increases as the total number of permits is increased. The delay is again the highest for the CS scheme, and the lowest for the CP scheme. However, the delay in all the three schemes follow a similar pattern.

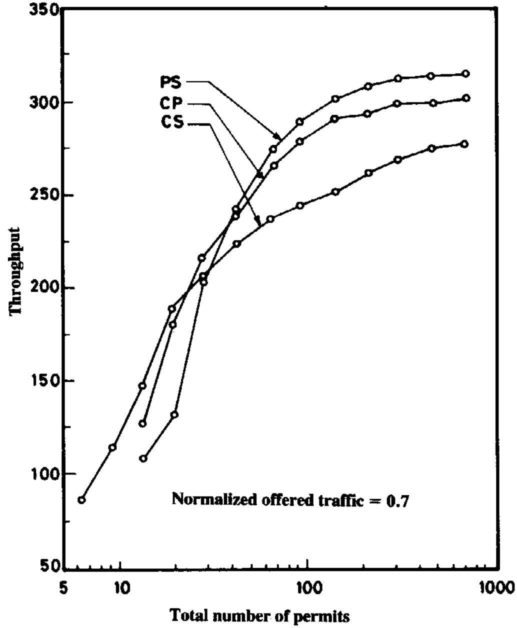

Fig. 7. Network throughput versus total number of permits

Figs. 9 and 10 probability of message rejection due to restrictions, versus the normalized offered traffic load and the total number of permits, respectively. These curves also cross over each other as the offered traffic load increases or the number of permits increases. However, for most of the range of the offered load and that for the number of permits, the PS scheme offers the lowest probability of rejection of a message.

Fig. 11 shows the histogram of the network transit delay per message versus percentage of messages for all the three permit distribution schemes. This histogram essentially shows the same characteristics which have been shown in previous results regarding the network transit delay. However, one thing should be noticed here that in the CS scheme a large percentage number of messages has a very small network delay and at the same time there is a relatively larger percentage of messages which has a very large delay. Also very few messages in this scheme have medium delay.

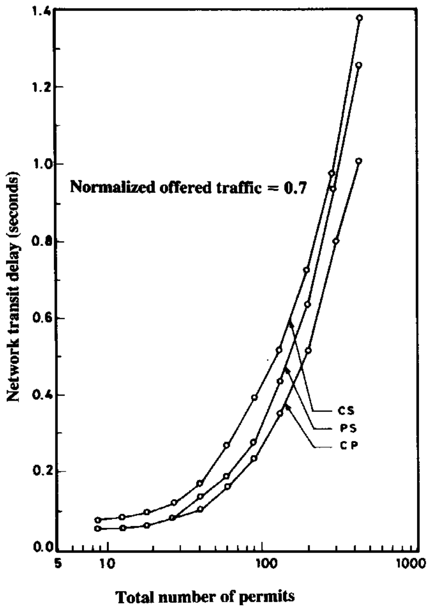

Fig. 8. Network transit delay per message versus total number of permits

This is mainly because of the nature of the scheme. Other two schemes follow a similar pattern.

Conclusions

In this paper we have considered three schemes of permit distribution in a global or isarithmic congestion avoidance method for computer communication networks. The basic idea behind these schemes is to prevent active users from monopolizing all the permits and at the same time prevent relatively less active nodes from starving for permits. The schemes discussed in this paper are complete sharing (CS), complete partitioning (CP), and proportional sharing (PS). These schemes have been simulated for a ten-node network and several illustrative simulation results are presented. It has been observed that the PS scheme has the best overall performance. This scheme also in the lowest probability of message rejection due to entry restrictions placed by the congestion avoidance schemes.

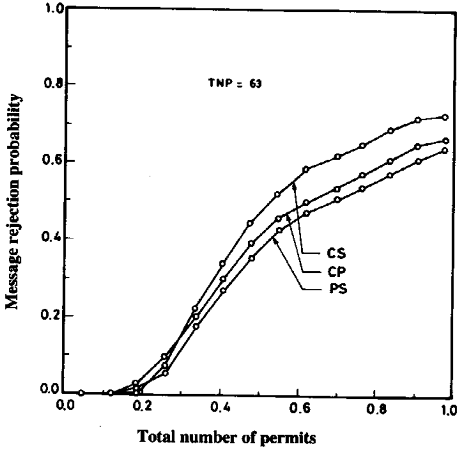

Fig. 9. Message rejection probability versus normalized offered traffic

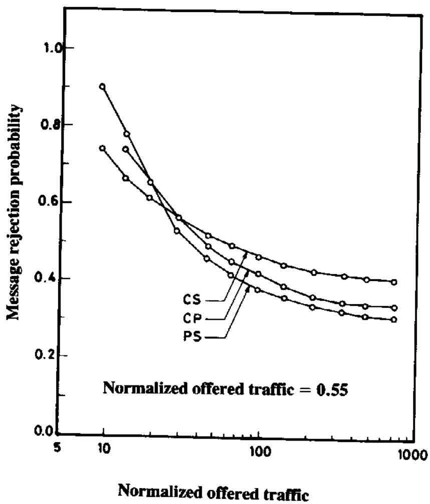

Fig. 10. Message rejection probability versus total number of permits

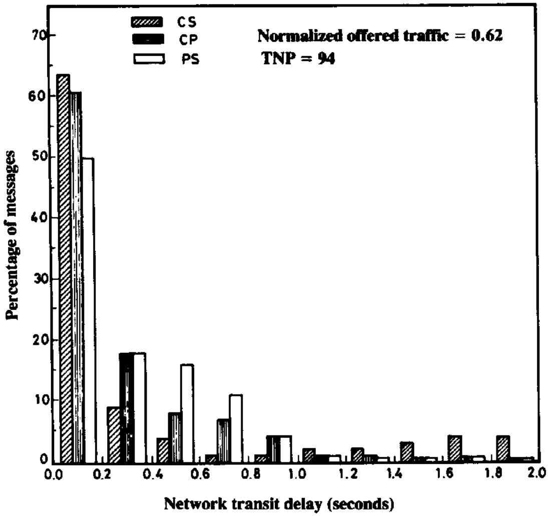

Fig. 11. Histogram of network transit delay per message

References

[1] Gerla, M. and Kleinrock, L. "Flow Control: A Comparative Survey." *IEEE Transactions on Communications,* Vol. COM-28, No. 4, April (1980), pp.553-574.

[2] Gerla, M. and Staskauskas, M. "Fairness in Flow Controlled Networks." *Journal of Telecommunication Networks,* 1 (1980), 19-30.

[3] Georganas, N.D. "Modelling and Analysis of Message Switched Computer Communication Networks with Multilevel Flow Control." *Computer Network,* 4, No.5 (1980) 285-294.

[4] Schwartz, M. *Computer Communication Networks: Design and Analysis.* NJ: Prentice-Hall, 1977.

[5] Wong, J.W. and Unsoy, M.S. "Analysis of Flow Control in Switched Data Networks." *Proc. of IFIP Conference,* August (1977).

[6] Kleinrock, L. *Queueing Systems: Vol. I and II.* New York: John Wiley, 1976.

[7] Ilyas, M. and Mouftah, H.T. "Performance Evaluation of Computer Communication Networks." *IEE Communications Magazine,* 23, No.4 (1985), 18-29.

[8] Kobayashi, H. "*Modeling and Analysis: An Introduction of System Performance Methodology.* London: Addison-Wesley Publishing Co., 1978.

[9] Shoemaker, S. *Computer Network and Simulation.*" Holland: North-Holland Publishing Company, 1978.

محاكاة تجنب منتظم للازدحام في شبكات الحاسوب

محمد إلياس

قسم هندسة الحاسوب، جامعة فلوريدا الأطلسية

ملخص البحث. يهتم هذا البحث بالتجنب العام (ويدعى أيضا بالمنتظم) للإزدحام في شبكات الحاسوب. ويعتمد هذا الأسلوب على الاحتفاظ بمستوى مناسب لحركة المعطيات داخل الشبكة، عن طريق فرض حد أعلى على عدد حزم المعطيات التي يمكن تقديمها للشبكة في أي وقت. وينفذ هذا الأسلوب باستخدام تصريحات سماح. فالرسالة التي تدخل الشبكة يجب أن تحصل على سماح، والرسالة التي تخرج من الشبكة تعيد السماح الذي حصلت عليه. يبين البحث أن أداء شبكة حاسوب تستخدم هذا الأسلوب الذي يعتمد على توزيع تصريحات السماح بين عُقد الشبكة. ويهتم البحث بثلاث طرق للتوزيع هي: التوزيع على أساس المشاركة الكاملة، التقسيم الكامل والمشاركة النسبية. وتكمن الفكرة الأساسية لهذه الطرق في منع المستخدم الفعّال من السيطرة على كل تصريحات السماح. ويقدم البحث مقارنة بين هذه الطرق. وتعطي نتائج المحاكاة: متوسط تأخير الرسالة، والإنتاجية. وتشير النتائج إلى أن طريقة المشاركة النسبية تعطي أداء عاما أفضل.